Int. To Deep Learning –HW#2

About the Assignment

The main aim of the assignment is to make you familiar with a traditional classifier by using KNN. Contributions of this lab are;

- Learning the K-NN classification method.
- Understanding the idea behind the classification task.

Submit the Assignment

Send your code and pdf document as zipped.

Ex: Name_Surname.zip

Hint

Look at the example in lecture notes.

Step1:

Read train images from train folder and convert images to vector format by writing a snipped code. It means that an image with 256x256x3 channels, must be converted to grayscale as 128x128 format. Then take derivative of image. Later, convert gradient of image to 1x16384 vector. Vector must be float16. Extract following 6 features from each of 1x16384 vector.

No.	Statistical Feature	Equation
1	Skewness Factor	$\frac{\frac{1}{N}\sum_{n=1}^{N}(x(n)-\bar{x})^3}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(x(n)-\bar{x})^2}\right)^3}$
2	Kurtosis Factor	$\frac{\frac{1}{N}\sum_{n=1}^{N}(x(n)-\bar{x})^4}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(x(n)-\bar{x})^2}\right)^4}$
3	Crest Factor	$\frac{\max\lvert x(n)\rvert}{\sqrt{\frac{1}{N}\sum_{n=1}^{N}x(n)^2}}$
4	Shape Factor	$\frac{\sqrt{\frac{1}{N}\sum_{n=1}^{N}x(n)^2}}{\frac{1}{N}\sum_{n=1}^{N}\lvert x(n)\rvert}$
5	Impulse Factor	$\frac{\max\lvert x(n)\rvert}{\frac{1}{N}\sum_{n=1}^{N}\lvert x(n)\rvert}$
6	Margin Factor	$\frac{\max\lvert x(n)\rvert}{\left(\frac{1}{N}\sum_{n=1}^{N}\sqrt{\lvert x(n)\rvert}\right)^2}$

Ref: Hoou, Hui Kar, Ooi Ching Sheng, Lim Meng Hee, and Leong Mohd Salman. "Feature selection tree for automated machinery fault diagnosis." In *MATEC Web of Conferences*, vol. 255, p. 02004. EDP Sciences, 2019.

There are 40 ***camera*** images, 53 ***flamingo*** images and 42 ***pizza*** images in **train folder**.

There are 4 ***camera*** images, 8 ***flamingo*** images and 6 ***pizza*** images in **test folder**.

In total you will have 135x6 train data, which is called as x_train.
Also y_train includes 135x1 label names.

Again, for each image, you will have 1x6 test data, which is called as ***sample_test***.

Task1:

Write a KNN function send the parameters of

```
def KNN(x_train, y_train, sample_test, k )
return 0
```

x_train holds image data related to ***camera (1)*** , ***flamingo (2)*** and ***pizza (3).***

y_train holds labels related to ***camera (1)*** , ***flamingo (2)*** and ***pizza (3).***

Sample_test is an image from test folder.

k is the nearest neighbor size and it is equal to 5.

The KNN function should return the most similar class name for sample_test. In case of computing the similarity, you are expected to use the Euclidean distance, which is explained in lecture notes by Teacher in class.